

LISTING OF CLAIMS:

Claim 1 (Previously Presented): A method for re-formatting computer files, comprising the steps:

inputting a data file into a computer having a specified operating system;

using said computer to determine if the data file is compatible with the specified operating system;

if the data file is not compatible with the computer, said computer transmitting the data file over the Internet from said computer to a universal server; and

the universal server, transforming the data file into a format compatible with the specified operating system of the computer, and sending the transformed data file back to the computer.

Claim 2 (Previously Presented): A method according to Claim 1, wherein the transforming step includes the steps of, the universal server reading information from said computer; and on the basis of said information, identifying the type of file, and transforming the file into a different format of the same type.

Claim 3 (Original): A method according to Claim 1, further comprising the steps of:

a user of the computer identifying user requirements; and

transmitting the user requirements to the universal server; and wherein

the transforming step includes the step of re-formatting the file in accordance with the user requirements.

Claim 4 (Original): A method according to Claim 1, wherein, when data needs to be converted, the data are sent to a universal conversion server; the universal conversion server checks user requirements; if the universal conversion server finds that the service cannot convert a certain file, the service looks in a computer description; the computer description can be located on the computer or on a universal conversion server database.

Claim 5 (Previously Presented): A method according to Claim 1, wherein the data file is a computer program, and the transforming step includes the step of the universal server looking over the program to identify components of the program including links to the program source code, the program's executable code, the program's file name; entering data to a database of source codes, where many source codes are held; and if the same name exists among more than one program in the database, the Universal Server reads the information from a description module.

Claim 6 (Cancelled).

Claim 7 (Previously Presented): A universal program conversion method, comprising the steps:

entering data into a computer;

said computer having a specified operating system and checking to determine whether the format of the data is compatible with the specified operating system(OS) in the computer;

if the format is not compatible, said computer sending the data from the computer over a network to a remote Universal Driver;

on the Universal Driver, reformatting the data into a format compatible to the specified OS;

after the reformatting step, sending the data to a universal formatting server, to be converted to a format suitable for the user;

if it is determined that the data are compatible with the operating system, then checking to determine whether it is necessary to reformat the data;

if the data do not need to be reformatted, processing the data as the user requests; and otherwise, sending the data to the universal server; and this server checking whether the data are executables;

if the data are executables, then checking the Universal Driver to determine whether the data can be formatted on the Universal Driver; if the data can be so formatted, then formatting the data at the Universal Driver; and then sending the formatted data to the user; if the data can not be formatted at the Universal Driver, then checking to determine if the source code exists on a storage of source code; if the source code exists, the Universal Driver then recompiling the data in a new OS, and the Universal Driver then sending the data to the user; checking for instructions to format data; after the checking step, formatting the data are formatted according to the instructions, and then sending the data to the user.

Claim 8 (Previously Presented): A system for re-formatting computer files, comprising:

a computer having input means for receiving a data file;

said computer including a specified operating system and means for determining if the data file is compatible with the specified operating system of the computer; and

a universal server for reformatting data;

said computer including means for transmitting the data file over the Internet from said computer to the universal server, if the data file is not compatible with the specified operating system of the computer; and

wherein the universal server includes means for transforming the data file into a format compatible with the specified operating system of the computer, and means for sending the transformed data file back to the computer.

Claim 9 (Original): A system according to Claim 8, wherein the transforming means includes means for identifying the type of file, and for transforming the file into a different format of the same type.

Claim 10 (Previously Presented): A system according to Claim 8, wherein a user of the computer identifies user requirements; and the system further comprises:

means for transmitting the user requirements to the universal server; and wherein

the transforming means includes means for re-formatting the file in accordance with the user requirements.

Claim 11 (Original): A system according to Claim 8, wherein, when data needs to be converted, the data are sent to a universal conversion server; the universal conversion server checks user requirements; if the universal conversion server finds that the service cannot convert a certain file, the service looks in a computer description; the computer description can be located on the computer or on a universal conversion server database.

Claim 12 (Previously Presented): A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for re-formatting computer files, the method steps comprising:

inputting a data file into a computer having a specified operating system;

using said computer to determine if the data file is compatible with the specified operating system of the computer;

if the data file is not compatible with the computer, ~~transmitting~~ using said computer to transmit the data file over the Internet from the computer to a universal server; and

the universal server, transforming the data file into a format compatible with the specified operating system of the computer, and sending the transformed data file back to the computer.

Claim 13 (Previously Presented): A program storage device according to Claim 12, wherein the transforming step includes the steps of, the universal server reading information from said computer; and on the basis of said information, identifying the type of file, and transforming the file into a different format of the same type.

Claim 14 (Original): A program storage device according to Claim 12, further comprising the steps of:

a user of the computer identifying user requirements; and

transmitting the user requirements to the universal server; and wherein

the transforming step includes the step of re-formatting the file in accordance with the user requirements.

Claim 15 (Original): A program storage device according to Claim 12, wherein, when data needs to be converted, the data are sent to a universal conversion server; the universal conversion server checks user requirements; if the universal conversion server finds that the service cannot convert a certain file, the service looks in a computer description; the computer description can be located on the computer or on a universal conversion server database.

Claim 16 (Previously Presented): A program storage device according to Claim 12, wherein the data file is a computer program, and the transforming step includes the step of the universal server looking over the program is looked over to identify components of the program including links to the program source code, the program's executable code, the program's file name; entering data to a database of source codes, where many source codes are held; and if the same name exists among more than one program, then the Universal Server reads the information from the description module.

Claim 17 (Previously Presented): A method according to Claim 1, further comprising the step of providing the Universal Server with access to a module having a series of source codes, and wherein the step of formatting the data file into a format compatible with the operating system of the computer modules includes the steps of:

the Universal Server obtaining from said module the source code for the data file; and

the Universal Server recompiling the data file, using the source code obtained from said module, into the format compatible with the operating system of the computer.

Claim 18 (Previously Presented): A method according to Claim 17, wherein the step of the Universal Server recompiling the data file includes the steps of, the Universal Server using the source code obtained from said module to modify the source code of the data file; and

using a compiler to compile a new data file, compatible with computer, from the modified source code of the data file.

Claim 19 (Previously Presented): A method according to Claim 18, further comprising the step of the Universal Server reading from the computer the type of operating system on the computer.

Claim 20 (Currently Amended): A method according to Claim 1, wherein the computer is of a given type, has a driver of a given type, includes a word processing application, and is operated by a user, the specified operating system is a first operating system, and the data file is a given program having a first source code, and comprising the further steps of:

providing first and second databases, using the first database to identify a group of user requirements, and using the second database to hold information about the computer;

the user entering into the first database information identifying

- i) the data file,
- ii) text that needs to be converted into a given word processing application,
and
- iii) files that need to be converted into a given audio application; ~~and~~
- ~~iv) compressed text that needs to be converted into post script format;~~

providing the second database with computer information identifying (i) the specified operating system, (ii) the type of the computer, (iii) the type of the driver of the computer, and (iv) the word processing application on the computer, including the steps of storing the second database on a universal conversion server database, and reading said computer information from a file on the computer;

wherein the step of transforming the data file includes the steps of

- i) using a universal driver to receive the data file from the computer,
- ii) the universal driver sending the data file to the universal server,
- iii) linking the universal server to a multitude of source codes, including the step of providing a table including links to source codes, the given program's executable code, and the given program's file name,
- iv) providing a license agreement identifying allowable compilations and fees for said compilations,

the universal server

- ~~v) checking the license agreement to determine if compilation of the given program into a second source code is allowable,~~
- vi) defining automatically what changes are needed to the given program and how the given program should be changed,
- ~~vii~~ vi) defining the type of data in the given program and the operating systems for which the given program was formatted,
- ~~viii~~ vii) checking to determine what operating system is on the computer and what applications are on the computer to process the type of data in the given program,
- ix ~~viii~~) sending the file data to a formatting server and asking the formatting server to reformat the file data to an application available on the computer,

- * ix) searching whether there is a source code that was used to compile the given program received from the computer,
- * x) using the first source code to compile a second source code for the specified operating system, including the step of adapting the first source code for compilation in the specified operating system; and after said first source code is adapted, compiling said second source code using a compiler from a set of compilers; and

wherein the step of sending the transformed data file back to the computer includes the step of sending said compiled second source code to the computer.

Claim 21 (Previously Presented): A method according to Claim 20, wherein:

the step of transforming the data file includes the further step of the universal server using the information read from the computer to determine how to change the data file received from the computer;

the step of linking the universal server to the multitude of source codes includes the step of storing said source codes in a module; and

the step of searching whether there is a source code that was used to compile the given program includes the step of searching said module for the first source code.